Matrizes

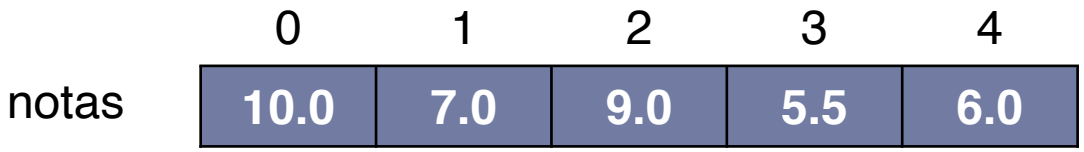
Loana Tito
loanatito@id.uff.br

Matrizes

- Variável composta **multidimensional**
  - É equivalente a um vetor, contudo permite a utilização de diversas dimensões acessadas via diferentes índices
  - Pode ser pensada como um vetor onde cada célula é outro vetor
  - Em diversas situações matrizes são necessárias para correlacionar informações

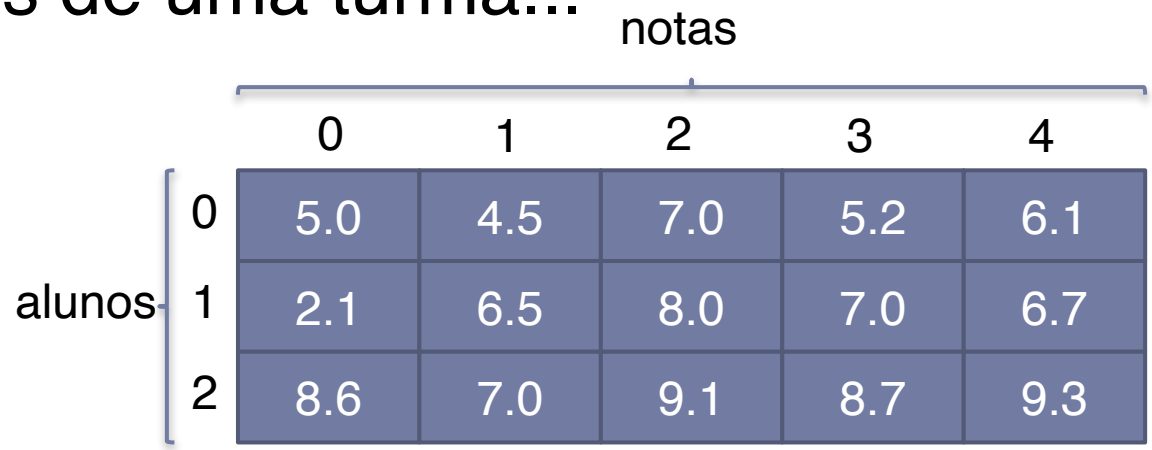
Exemplo motivacional

- Assumindo que **um aluno é avaliado com cinco notas**, seria necessário um vetor de cinco posições para guardar as notas de um aluno...



Exemplo motivacional

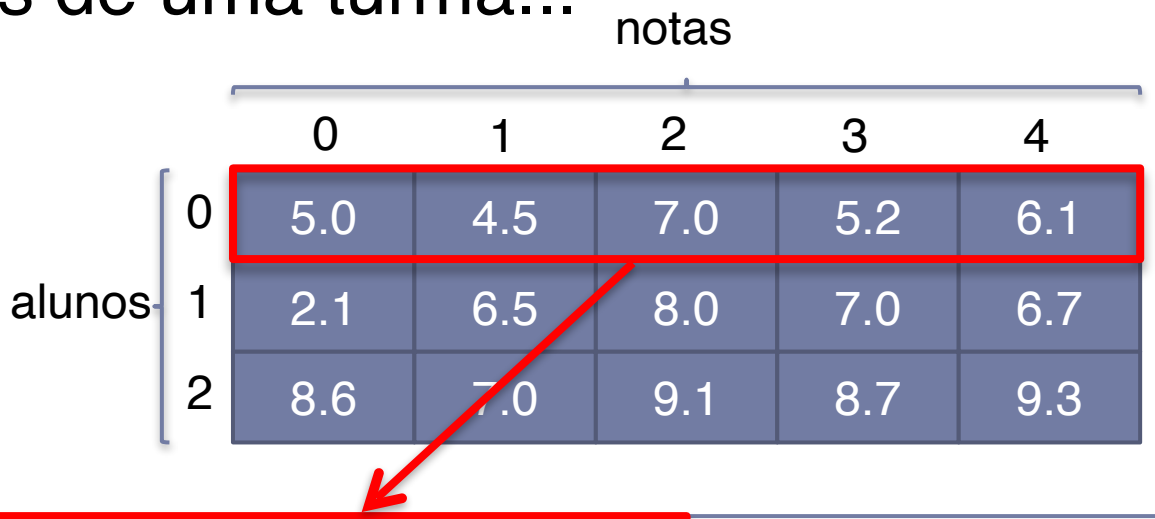
- Contudo, assumindo que **uma turma tem três alunos**, seria necessária uma matriz bidimensional para guardar as notas de todos os alunos de uma turma...




```
notas = [[5.0, 4.5, 7.0, 5.2, 6.1], [2.1, 6.5, 8.0, 7.0, 6.7],
    [8.6, 7.0, 9.1, 8.7, 9.3]]
```

Exemplo motivacional

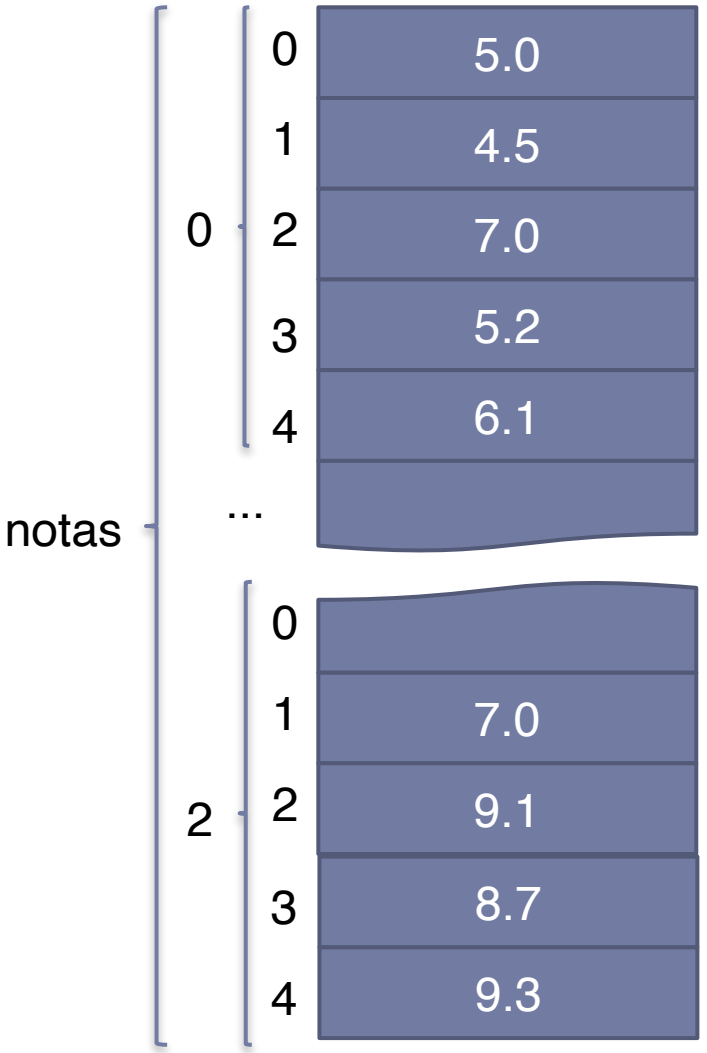
- Contudo, assumindo que **uma turma tem três alunos**, seria necessária uma matriz bidimensional para guardar as notas de todos os alunos de uma turma...




```
notas = [[5.0, 4.5, 7.0, 5.2, 6.1], [2.1, 6.5, 8.0, 7.0, 6.7],
    [8.6, 7.0, 9.1, 8.7, 9.3]]
```

Exemplo motivacional

- Na verdade, na memória seria algo assim...



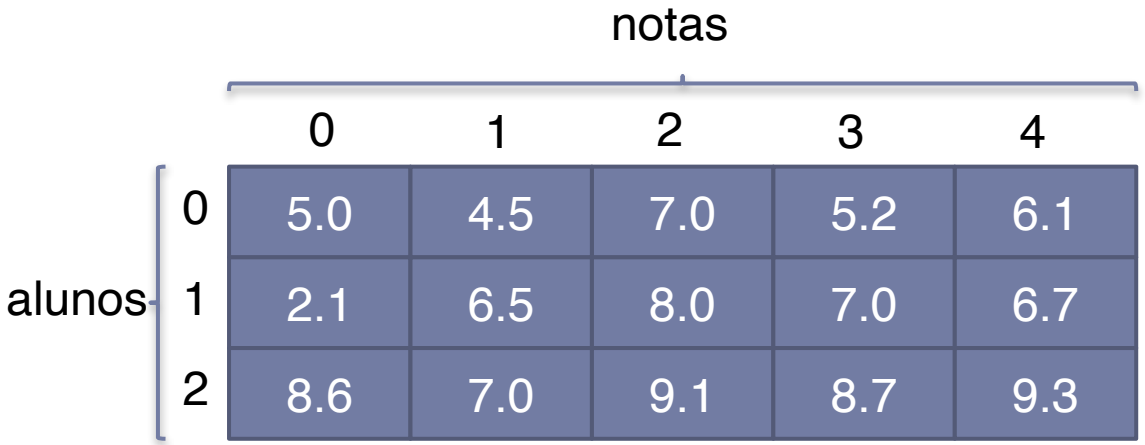
Acesso aos valores: [linha][coluna]

- Segunda nota do primeiro aluno

```
>>> notas[0][1]
4.5
```

- Quinta nota do terceiro aluno

```
>>> notas[2][4]
9.3
```



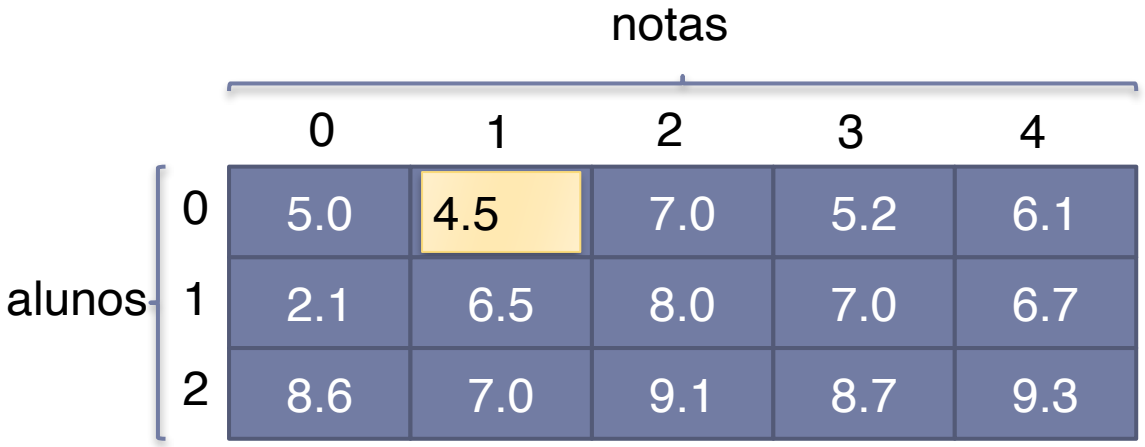
Acesso aos valores: [linha][coluna]

- Segunda nota do primeiro aluno

```
>>> notas[0][1]
4.5
```

- Quinta nota do terceiro aluno

```
>>> notas[2][4]
9.3
```



Calcular a média da turma

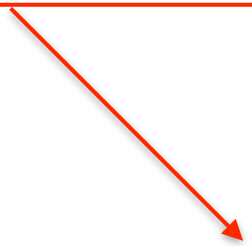
```
notas = [[5.0, 4.5, 7.0, 5.2, 6.1], [2.1, 6.5, 8.0, 7.0, 6.7], [8.6, 7.0, 9.1, 8.7, 9.3]]
#calcula a média
soma = 0
#for para percorrer as linhas
for i in range(3):
    #for para percorrer as colunas
    for j in range(5):
        soma = soma + notas[i][j]
media = soma/ 15
print(media)
```

Preencher a matriz por leitura

```
notas = []
for i in range(3):
    # cria linha vazia
    linha = []
    for j in range(5):
        #vai adicionando as notas na linha
        linha.append(eval(input('Digite a nota
[' + str(i) + ',' + str(j) + ']:')))
    #adiciona a linha na matriz turma
    notas.append(linha)
```

Preencher a matriz por leitura

```
notas = []
for i in range(3):
    # cria linha vazia
    linha = []
    for j in range(5):
        #vai adicionando as notas na linha
        linha.append(eval(input('Digite a nota
[' + str(i) + ',' + str(j) + ']:')))
    #adiciona a linha na matriz turma
    notas.append(linha)
```



Cria um vertor com as 5 notas de um aluno

Preencher a matriz por leitura

```
notas = []
for i in range(3):
    # cria linha vazia
    linha = []
    for j in range(5):
        #vai adicionando as notas na linha
        linha.append(eval(input('Digite a nota
[' + str(i) + ',' + str(j) + ']:')))
    #adiciona a linha na matriz turma
    notas.append(linha)
```

Repete esse procedimento 3 vezes, adicionando cada linha com as 5 notas à matriz NOTAS.

Exemplo

- Programa que cria uma matriz **n x m** preenchida com zeros

```
n = eval(input('Digite a dimensão n da matriz: '))
m = eval(input('Digite a dimensão m da matriz: '))
matriz = []
for i in range(n):
    linha = []
    for j in range(m):
        linha.append(0)
    matriz.append(linha)
print(matriz)
```

Simplificando o exemplo

- Programa que cria uma matriz **n x m** preenchida com zeros

```
n = eval(input('Digite a dimensão n da matriz: '))
m = eval(input('Digite a dimensão m da matriz: '))
matriz = []
for i in range(n):
    matriz.append([0]*m)
print(matriz)
```

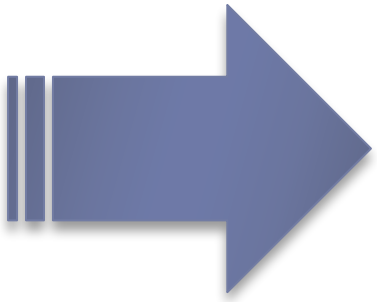
Imprimir em forma de matriz

- Programa que cria uma matriz **n x m** preenchida com zeros **e a imprime no formato de matriz**

```
n = eval(input('Digite a dimensão n da matriz: '))
m = eval(input('Digite a dimensão m da matriz: '))
matriz = []
for i in range(n):
    matriz.append([0]*m)
#imprimir em formato de matriz
for i in range(n):
    print(matriz[i])
```

Exemplo Contar Pares

- Programa que lê uma matriz 3x3 digitada pelo usuário e conta quantos números pares existem na matriz, imprimindo na tela o resultado e a matriz.

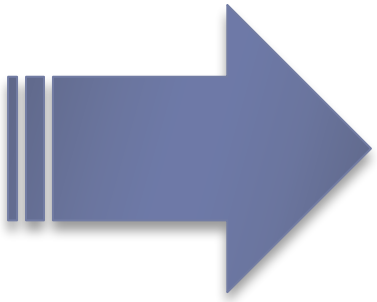


Exemplo Contar Pares

```
matriz = []
for i in range(3):
    linha = []
    for j in range(3):
        linha.append(input('Digite o valor de [' +
str(i) + ',' + str(j) + ']:'))
    matriz.append(linha)
#contar pares
pares = 0
for i in range(3):
    for j in range(3):
        if matriz[i][j] % 2 == 0:
            pares += 1
#imprimir em formato de matriz
for i in range(3):
    print(matriz[i])
#imprimir qtde de números pares
print('A matriz contém', pares, 'números pares')
```

Python permite misturar tipos em uma matriz

- Exemplo: programa que armazena os nomes e idades de 10 pessoas em uma matriz, e imprime o nome da pessoa mais nova

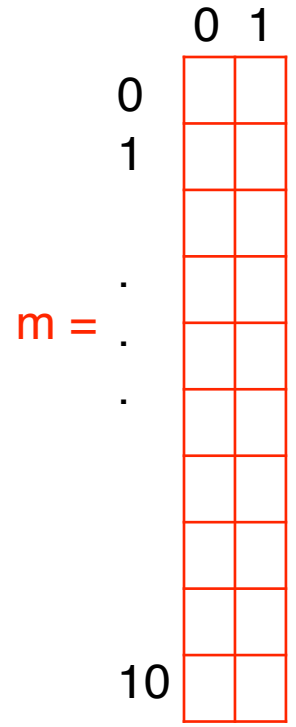


Encontra a pessoa mais nova

```
m = []
#preenche a matriz
for i in range(10):
    linha = []
    linha.append(input('Digite o nome da pessoa ' +
str(i) + ':'))
    linha.append(int(input('Digite a idade de ' +
linha[0] + ':')))
    m.append(linha)
#procura a pessoa mais nova
menor = m[0][1]
pos = 0
for i in range(10):
    if m[i][1] < menor:
        menor = m[i][1]
        pos = i
#imprime a matriz
for i in range(10):
    print(m[i])
print('A pessoa mais nova é', m[pos][0])
```

Encontra a pessoa mais nova

```
m = []
#preenche a matriz
for i in range(10):
    linha = []
    linha.append(input('Digite o nome da pessoa ' +
str(i) + ':'))
    linha.append(int(input('Digite a idade de ' +
linha[0] + ':')))
    m.append(linha)
#procura a pessoa mais nova
menor = m[0][1]
pos = 0
for i in range(10):
    if m[i][1] < menor:
        menor = m[i][1]
        pos = i
#imprime a matriz
for i in range(10):
    print(m[i])
print('A pessoa mais nova é', m[pos][0])
```

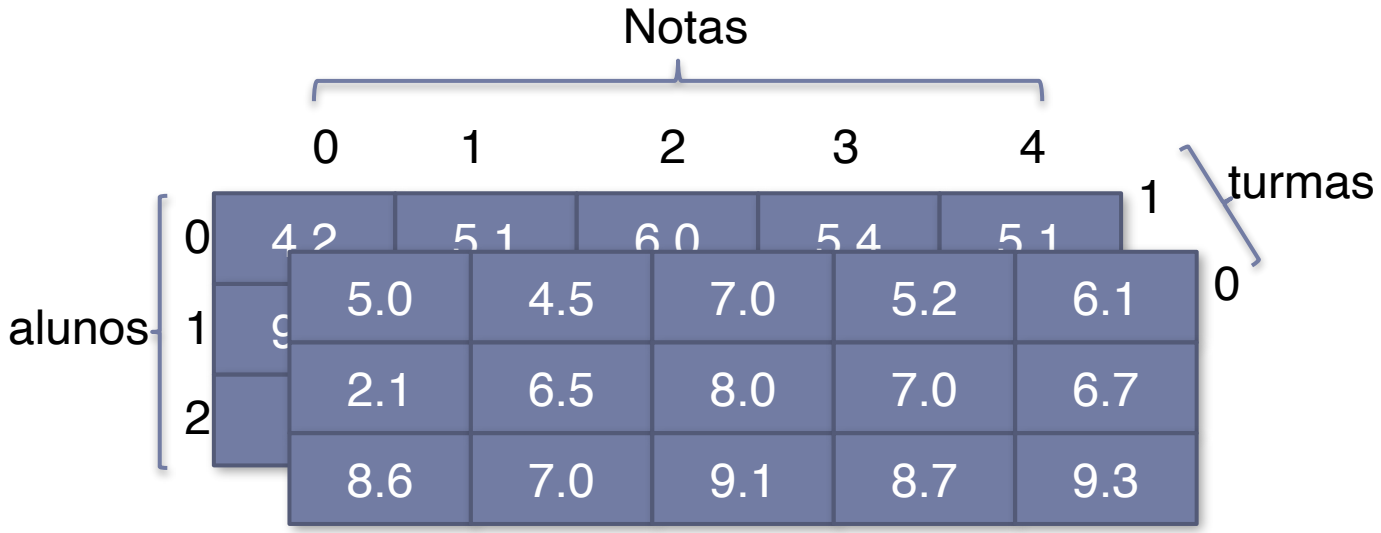


Matrizes

- Uma matriz pode ter um número qualquer de dimensões! Basta usar um índice para cada dimensão.

Exemplo motivacional

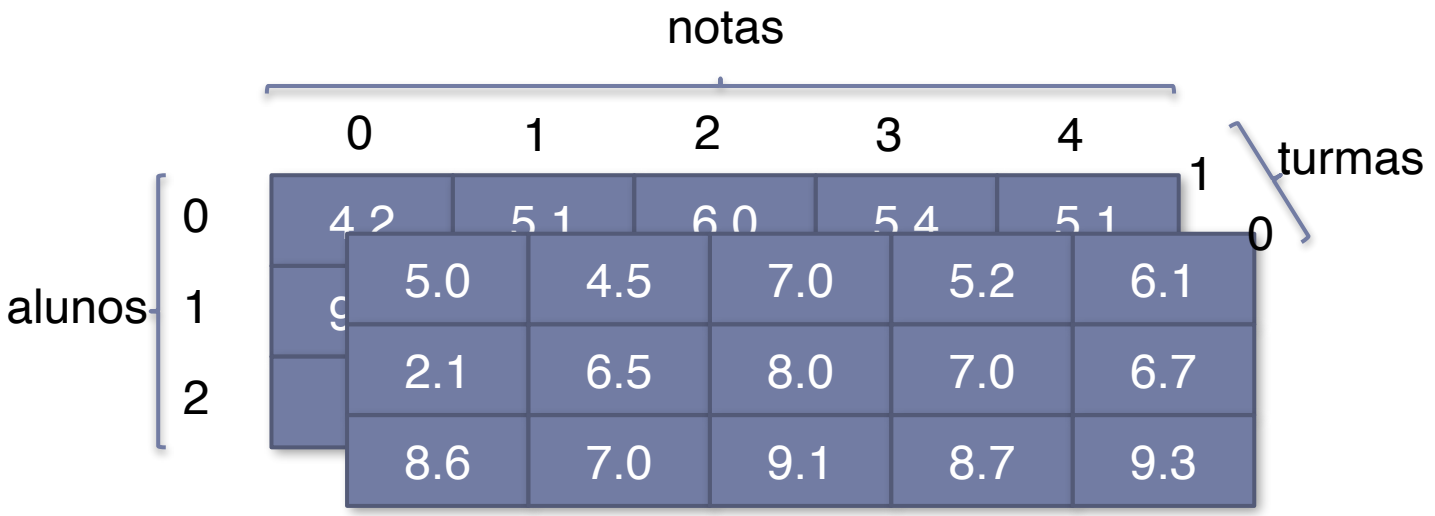
- Ainda, assumindo que **um curso tem duas turmas**, seria necessária uma matriz tridimensional para guardar as notas de todos os alunos de todas as turmas do curso.



Atribuição

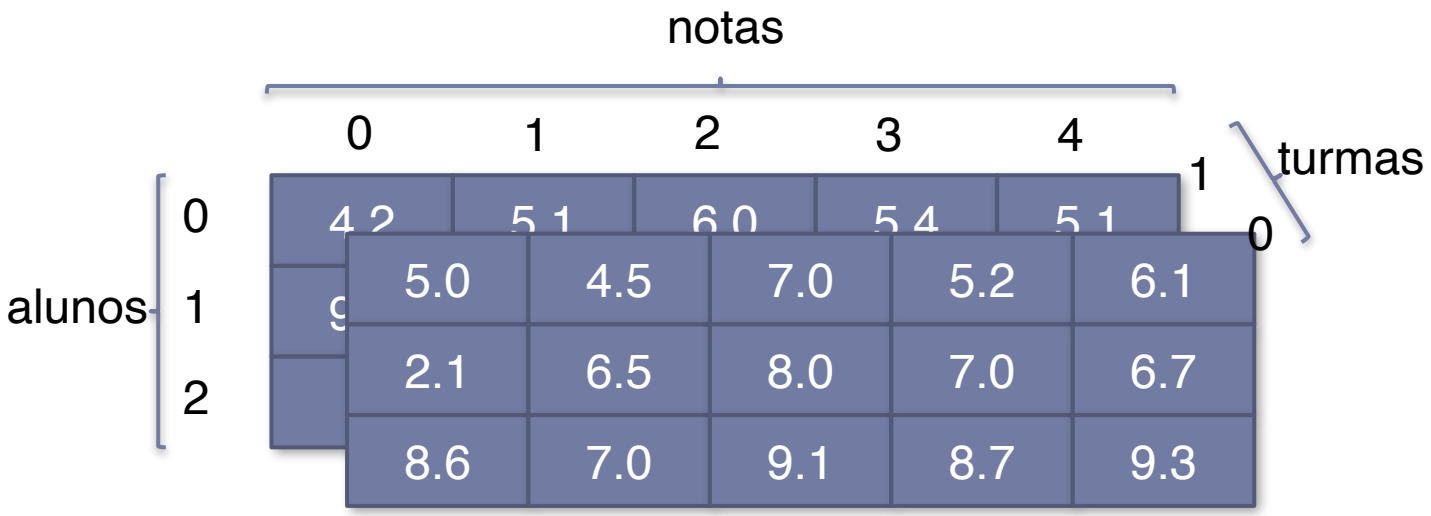
```
>>> notas = [[[5.0, 4.5, 7.0, 5.2, 5.1],
[2.1,6.5,8.0,7.0,6.7],[8.6,7.0,9.1,8.7,9.3]],
[[4.2,5.1,6.0,5.4,5.1],[9.0,8.0,7.5,8.1,8.8],
[2.3,4.4,6.7,6.6,7.0]]]
```

Turma 0



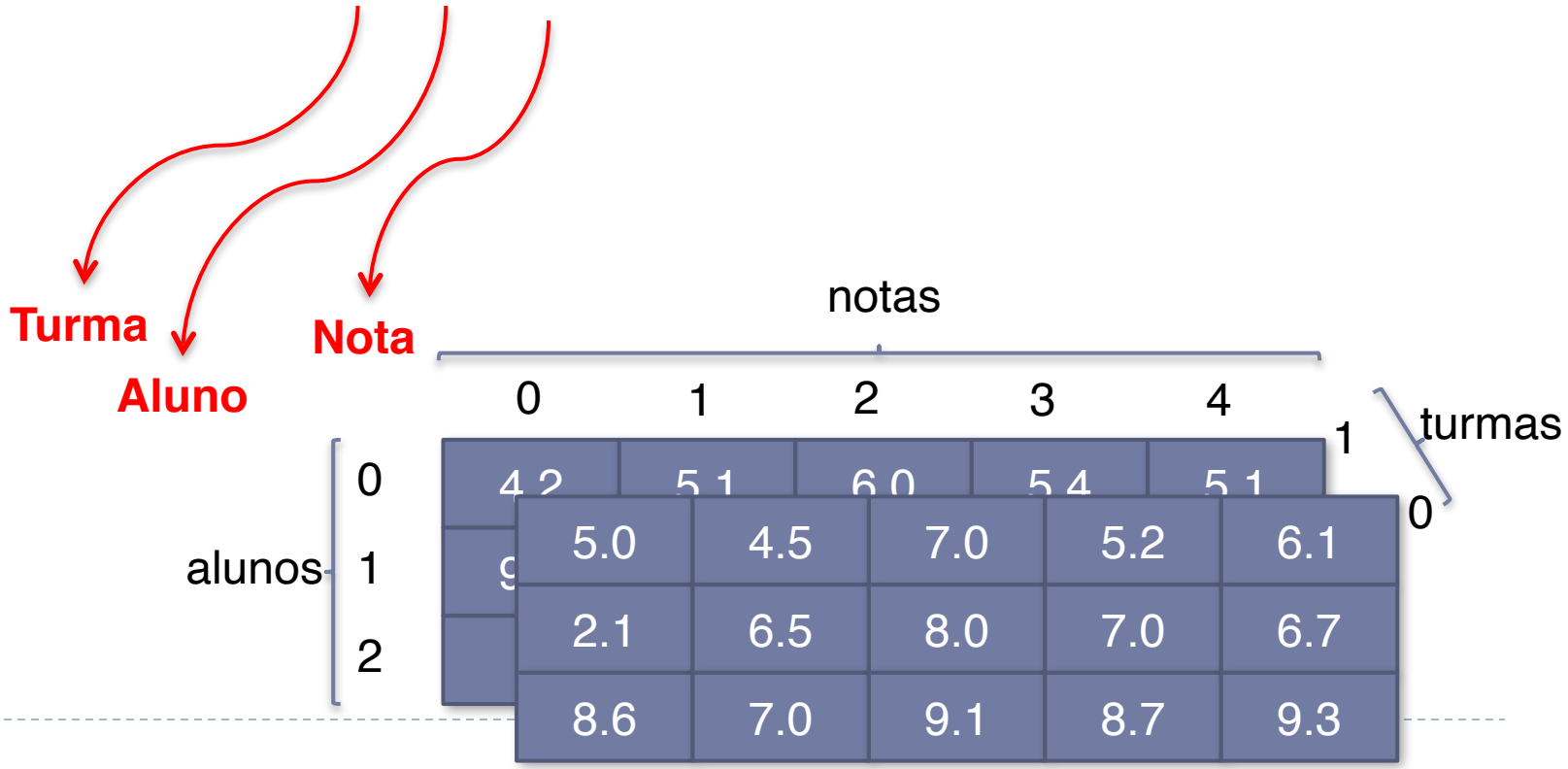
```
>>> notas = [[[5.0, 4.5, 7.0, 5.2, 5.1],
[2.1,6.5,8.0,7.0,6.7],[8.6,7.0,9.1,8.7,9.3]],
[[4.2,5.1,6.0,5.4,5.1],[9.0,8.0,7.5,8.1,8.8],
[2.3,4.4,6.7,6.6,7.0]]]
```

Aluno 0 da turma 0



Acesso a elemento

```
>>> notas = [[[5.0, 4.5, 7.0, 5.2, 5.1],
[2.1,6.5,8.0,7.0,6.7],[8.6,7.0,9.1,8.7,9.3]],
[[4.2,5.1,6.0,5.4,5.1],[9.0,8.0,7.5,8.1,8.8],
[2.3,4.4,6.7,6.6,7.0]]]
>>> print(notas[0][1][0])
2.1
```



Exercícios

1. Faça um programa que leia uma matriz 3x3 e multiplique os elementos da diagonal principal da matriz por um número k. Imprima a matriz na tela antes e depois da multiplicação.
2. Faça um programa que leia duas matrizes A e B 2x2 e imprima a matriz C que é a soma das matrizes A e B.

Exercícios

4. Faça um programa que leia uma matriz 3x3 de inteiros e retorne a linha de maior soma. Imprima na tela a matriz, a linha de maior soma e a soma.
5. Faça um programa que leia a ordem de uma matriz quadrada A (até 100), posteriormente leia seus valores e escreva sua transposta AT, onde AT[i][j] = A[j][i].
6. Faça um programa que leia uma matriz 6x3 com números reais, calcule e mostre: (a) o maior elemento da matriz e sua respectiva posição (linha e coluna); (b) o menor elemento da matriz e sua respectiva posição.

Referências

- Slides preparados em conjunto com Vanessa Braganholo e Aline Paes

Matrizes

Leonardo Murta
leomurta@ic.uff.br